



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

## ÚSTAV TELEKOMUNIKACÍ

DEPARTMENT OF TELECOMMUNICATIONS

## NÍZKOPŘÍKONOVÝ BEZDRÁTOVÝ SENZOROVÝ UZEL

LOW-POWER WIRELESS SENSOR NODE

### BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

### AUTOR PRÁCE

AUTHOR

Michal Pobiecky

### VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Ondřej Krajsa, Ph.D.

BRNO 2017

# Bakalářská práce

bakalářský studijní obor **Teleinformatika**

Ústav telekomunikací

**Student:** Michal Pobiecky

**ID:** 173727

**Ročník:** 3

**Akademický rok:** 2016/17

**NÁZEV TÉMATU:**

## Nízkopříkonový bezdrátový senzorový uzel

### POKYNY PRO VYPRACOVÁNÍ:

V rámci bakalářské práce navrhnete a realizujete senzorový uzel využívající linkovou vrstvu IEEE 802.15.4. Návrh proveďte se snahou o minimální spotřebu energie a minimální rozměry zařízení.

### DOPORUČENÁ LITERATURA:

[1] OBAIDAT, Mohammad S. a Sudip MISRA. Inside a wireless sensor node: structure and operations. Principles of Wireless Sensor Networks. Cambridge: Cambridge University Press, 2012, s. 14-29. DOI: 10.1017/CBO9781139030960.003. ISBN 9781139030960.

[2] SINCLAIR, Ian R. Practical Electronics Handbook. 3rd ed. Kent: Elsevier Science, 2016. ISBN 9780750606912.

**Termín zadání:** 1.2.2017

**Termín odevzdání:** 8.6.2017

**Vedoucí práce:** Ing. Ondřej Krajsa, Ph.D.

**Konzultant:**

**doc. Ing. Jiří Mišurec, CSc.**  
předseda oborové rady

### UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

# ABSTRAKT

Cieľom tejto práce je navrhnúť a vytvoriť bezdrôtový senzorový uzol s dôrazom na nízku spotrebu energie. Prvotne je toto zariadenie postavené na mikrokontroléri Atmel ATmega328P a namerané dáta odosiela pomocou XBee využívajúc vrstvu IEE 802.15.4., prípadne modul nRF24L01. Následne návrh využíva Atmel ATmega256RFR2 s integrovanou bezdrôtovou časťou a protokolom Lightweight Mesh. Špeciálna pozornosť je v tejto práci kladená najmä na režimy spánku a znižovanie spotreby mikrokontroléra. Tieto hodnoty sú pre ATmega328P aj namerané.

## Kľúčové slová

Atmel ATmega256RFR2, Atmel ATmega328P, ISL29020, Lightweight Mesh, nízka spotreba, nRF24L01, režimy spánku, senzorový uzol, TMP102, XBee,

# ABSTRACT

The main purpose of this thesis is to design wireless sensor node with emphasis on low power consumption. This device is firstly made of microcontroller Atmel ATmega328P and for transferring measured data XBee 802.15.4 or transceiver nRF24L01 is used. Subsequently there is a design with Atmel ATmega256RFR2 with integrated wireless part and Lightweight Mesh protocol. Special attention is given to sleep modes and lowering power consumption. For ATmega328P the measurement of these power modes is done.

## KEYWORDS

Atmel ATmega328P, Atmel ATmega256RFR2, ISL29020, Lightweight Mesh, low power consumption, nRF24L01, sensor node, sleep modes, TMP102, XBee

POBIECKY, M. Nízkopříkonový bezdrátový senzorový uzel. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2017. 47 s. Vedoucí bakalářské práce Ing. Ondřej Krajsa, Ph.D..

# PROHLÁŠENÍ

Prohlašuji, že svou závěrečnou práci na téma „Nízkopříkonový bezdrátový senzorový uzel“ jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno, 08.06.2017

podpis autora

Výzkum popsaný v této diplomové práci byl realizovaný v laboratořích podpořených projektem Centrum senzorických, informačních a komunikačních systémů (SIX); registrační číslo CZ.1.05/2.1.00/03.0072, operačního programu Výzkum a vývoj pro inovace.

# **POĎAKOVANIE**

Rád by som poďakoval vedúcemu bakalárskej práce pánovi Ing. Ondřeji Krajsovi, Ph.D. za odborné vedenie, konzultácie, trpezlivosť a podnetné návrhy k práci.

Brno 08.06.2017

podpis autora

# OBSAH

<b>Úvod.....</b>	<b>12</b>
<b>1 Arduino UNO .....</b>	<b>14</b>
<b>2 Mikrokontrolér Atmel ATmega328P .....</b>	<b>16</b>
2.1 picoPower.....	16
2.2 Vypínanie perifériei .....	18
<b>3 Odosielanie dát s ATmega328P .....</b>	<b>19</b>
3.1 XBee.....	19
3.1.1 Uspávanie XBee.....	19
3.2 nRF24L01 2.4GHz .....	21
3.2.1 Dosah.....	21
<b>4 Režimy úspory a uspania.....</b>	<b>22</b>
4.1 Idle Mode .....	16
4.2 ADC Noise Reduction Mode .....	18
4.3 Power-Down Mode .....	16
4.4 Power-Save Mode .....	18
4.5 Standby Mode .....	16
4.6 Extended Standby Mode .....	18
<b>5 Meranie s ATmega328P.....</b>	<b>24</b>
5.1 Režimy optimalizácie spotreby .....	24
5.1.1 Power-down režim .....	25
5.1.2 Vypnutie AD prevodníka .....	26
5.1.3 Vypnutie BOD .....	26
5.1.4 Využitie WDT .....	28
<b>6 Mikrokontrolér Atmel ATmega256RFR2 .....</b>	<b>30</b>



6.1 Atmel Lightweight Mesh .....	31
<b>7 Senzory s ATmega256RFR2.....</b>	<b>33</b>
7.1 Teplotný senzor TMP102.....	33
7.2 Svetelný senzor ISL29020 .....	34
<b>8 Cena zariadenia s ATmega256RFR2 .....</b>	<b>35</b>
<b>Záver.....</b>	<b>37</b>
<b>Literatúra.....</b>	<b>39</b>
<b>Zoznam skratiek.....</b>	<b>42</b>
<b>Zoznam príloh .....</b>	<b>43</b>

# ZOZNAM OBRÁZKOV

Obr. 1.0: Graf vyhľadávania pojmu „Internet of things“ .....	13
Obr. 1.1: Popis Arduino UNO .....	14
Obr. 1.2: Kód základného testu .....	15
Obr. 2.1: Závislosť napájania na frekvencii mikrokontroléra .....	16
Obr. 2.2: Spotreba prúdu mikrokontrolérom pri rôznom napájacom napätí a teplote....	17
Obr. 2.3: Zapojenie tranzistora MOSFET pre ovládanie napájania periférie.....	18
Obr. 3.1: Modul XBee .....	19
Obr. 3.2: Zapojenie modulu XBee s funkciou uspávania .....	20
Obr. 3.3: Paket odosielaný XBee.....	20
Obr. 3.4: Príklad kódu pre uspanie a zobudenie XBee .....	21
Obr. 3.5: Zapojenie nRF24L01 s ATmega328P .....	21
Obr. 5.1: Základné zapojenie mikrokontroléra .....	24
Obr. 5.2: Základný kód bliknutia LED diódy .....	25
Obr. 5.3: SMCR – kontrolný register .....	25
Obr. 5.4: Zdrojový kód s využitím hlbokého spánku .....	26
Obr. 5.5: Register ADCSRA.....	26
Obr. 5.6: Kontrolný register – MCUCR .....	27
Obr. 5.7: Zdrojový kód s použitím vypnutia AD prevodníka a BOD .....	27
Obr. 5.8: Register WDTCSR .....	28
Obr. 5.9: Zdrojový kód s uplatnením vzbudenia WDT .....	29
Obr. 6.1: Spotreba mikrokontroléra a bezdrôtovej časti ATmega256RFR2 .....	30
Obr. 6.2: Architektúra Lightweight Mesh .....	31

Obr. 7.1: Odchýlka teploty v merateľnom spektre .....	33
-------------------------------------------------------	----

# ÚVOD

Táto práca sa bude venovať návrhu univerzálneho senzorového uzla postaveného na mikrokontroléri ATmega328P a následne ATmega256RFR2. Dáta budú získavané zo senzorov (teplota, svetlo... ) a následne odosielané pomocou modulu XBee, nRF24L01, respektíve bude využívaná vstavaná bezdrôtová časť mikrokontroléra ATmega256RFR2. Podstatou zariadenia je poskytnúť vybrané informácie z prostredia, v ktorom sa nachádza, bráne, ktorá ich spracuje, vyhodnotí a následne môže vykonať požadované akcie. Tento celok by mal byť prínosom pre rozvíjajúci sa fenomén Internetu vecí a prispieť k automatizácii procesov ako v priemyselnej sfére, tak i v oblasti inteligentných domácností.

Jedným z cieľov mojej bakalárskej práce je dôraz na nízku spotrebu energie, ktorý je kľúčový v tejto sfére. V ideálnom prípade by malo takéto zariadenie fungovať po čo najdlhšiu dobu. Pri konštrukcii prototypu sa preto zväčša používa nízkopríkonový mikrokontrolér a zariadenie sa uspáva.

## História IoT

Ako jeden z najstarších „inteligentných prístrojov“ môžeme považovať hriankovač, ktorý sa dokázal zapnúť a vypnúť pomocou Internetu. Bol pripojený k počítaču pomocou TCP/IP sieťovania a používal SNMP MIB a prepečenie bolo určované dĺžkou zapnutia hriankovača. Podarilo sa to v roku 1990 Američanovi menom John Romkey. To určilo smer, ktorým sa o niekoľko rokov začali zariadenia uberať. Následne bolo publikovaných mnoho článkov, ktoré predpovedali vzostup interakcie fyzického sveta medzi sebou a hromadné rozšírenie. [1]

Následne v roku 1999 bol pojem „Internet vecí“ zadefinovaný Kevinom Ashtonom, riaditeľom spoločnosti Auto-ID. Oficiálnou definíciu sa jedná o: „Senzory a ovládacie prvky implementované vo fyzických predmetoch spojené pomocou drôtovej alebo bezdrôtovej siete“. Prvotne bolo cieľom rozšíriť technológiu RFID. V roku 2000 LG uviedlo svoje prvé plány na zostrojenie inteligentnej chladničky. [2]

Zaujímavý je aj pohľad na popularitu pojmu „Internet of Things“ v jednom z najrozšírejších vyhľadávačov Google, ktorý ako sa zdá, odzrkadľuje popularnosť tohoto fenoménu.



Obr. 1.0: Graf vyhľadávania pojmu „Internet of things“ [3]

Z grafu je jasne pozorovateľné, že záujem o Internet vecí vzrástol najmä v období rokov 2010 a 2011, kedy aj čínsky premiér Wen Jiabao avizoval, že táto oblasť je pre čínsky priemysel kľúčová a plánujú ju významne podporiť. Veľkú rolu hrá bezpochyby aj verejné zavedenie nového protokolu IPv6, ktoré prinieslo  $2^{128}$  (približne 340 sextiliónov) nových adries. Následný verejný rozmach nastal na začiatku roku 2014. [4]

V dnešnej dobe by sme mohli Internet vecí rozdeliť na niekoľkých kategórií:

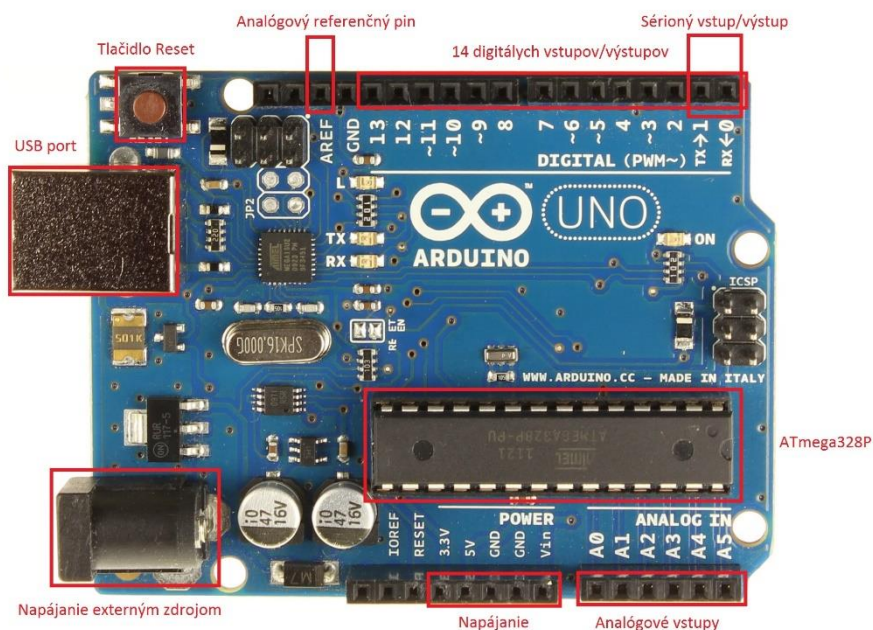
- nositeľné zariadenia / tréningové pomôcky
- automatizácia domácností
- priemyselná automatizácia

# 1 ARDUINO UNO

Arduino UNO je mikrokontrolér založený právě na spomínanom ATmega328P. Na náš prototyp by bol vhodný najmä kvôli jeho kompatibilita naprieč celým spektrom senzorov a rôznych zariadení. Medzi jeho výhody patrí taktiež široká podpora komunity (množstvo rozličných knižníc) a možnosť programovania pomocou prostredia „Arduino IDE“.

Tab. 1: Technická špecifikácia Arduino UNO [5]

Mikrokontrolér	ATmega328P
Prevádzkové napätie	5V
Vstupné napätie (odporúčané)	7-12V
Vstupné napätie (hraničné)	6-20V
Digitálne vstupy/výstupy	14 (6 z nich poskytuje PWM výstup)
PWM výstupy	6
Analogové vstupy	6
Maximálny prúd cez jeden pin	20 mA
Maximálny prúd z 3.3 V výstupu	50 mA
Flash pamäť	32 KB (ATmega328P) 0.5 KB využíva bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Frekvencia hodín	16 MHz
Dĺžka	68.6 mm
Šírka	53.4 mm
Hmotnosť	25 g



Obr. 1.1: Popis Arduino UNO [6]

Problémom však môže byť vysoká spotreba energie celého Arduina, kvôli napájaniu častí, ktoré reálne nevyužívame a preto sme to overili prakticky.

Pre základný test bude použitý kód veľmi jednoduchý:

```
1 void setup() {}  
2  
3 void loop() {}
```

Obr. 1.2: Kód základného testu

Pri zdroji napätia 5 V pripojeného na vstup Arduino UNO, spotrebovaný elektrický prúd činí 48.2 mA. Ak by sme spočítali výkon zariadenia podľa vzorca (1.1) pre výkon (W):

$$P = U \times I \text{ [W]} \quad (1.1)$$

po dosadení hodnôt nám vyjde hodnota výkonu:

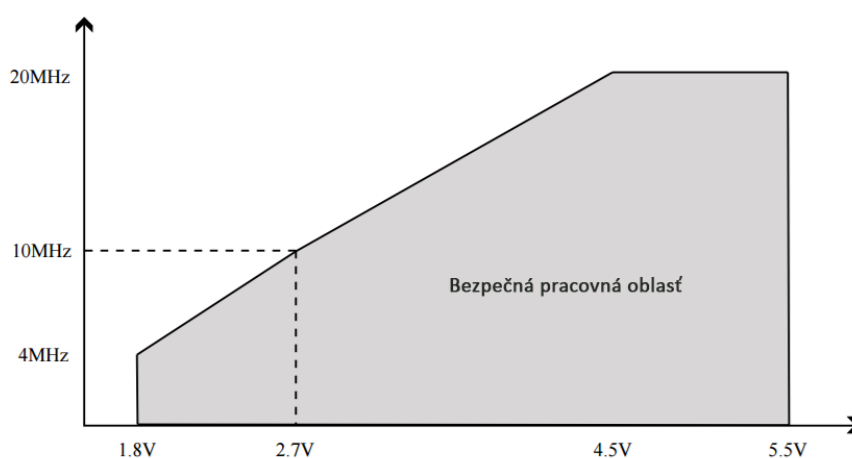
$$P = 5 \times 0,0482 \text{ [W]}$$

$$P = 0,241 \text{ W}$$

Táto hodnota je pre naše použitie príliš vysoká, batéria by nám toto zariadenie dokázala napájať len približne mesiac, preto si postavíme vlastný návrh zariadenia.

## 2 MIKROKONTROLÉR ATMEL ATMEGA328P

Tento 8 bitový mikrokontrolér Atmel ATmega328 je vo verzii P, čo znamená vyhotovenie picoPower. Je veľmi známy pre svoje široké použitie najmä v zariadeniach založených na báze Arduina. Preto naša prvotná voľba padla práve naň. Táto verzia nášho senzorového uzla bude mať teda rovnaké výhody ako Arduino, no jeho spotreba bude minimalizovaná. V spojení s vysielačom dát budeme vedieť vyčítať dáta zo senzorov, odoslať ich a následne mikrokontrolér ATmega328P uspať.



Obr. 2.1: Závislosť napájania na frekvencii mikrokontroléra

Z uvedeného grafu vyplývajú tieto údaje:

- napätie pod 1.8 V – mikrokontrolér nemôže fungovať
- frekvencia pod 4 MHz – napätie musí byť minimálne 1.8 V
- rozpätia medzi 4 MHz až 10 MHz a 10 MHz až 20 MHz sú lineárne

Napájacie napätie je podľa tabuliek od 1.8 V pri 4 MHz do 5.5 V pri 20 MHz. [7]

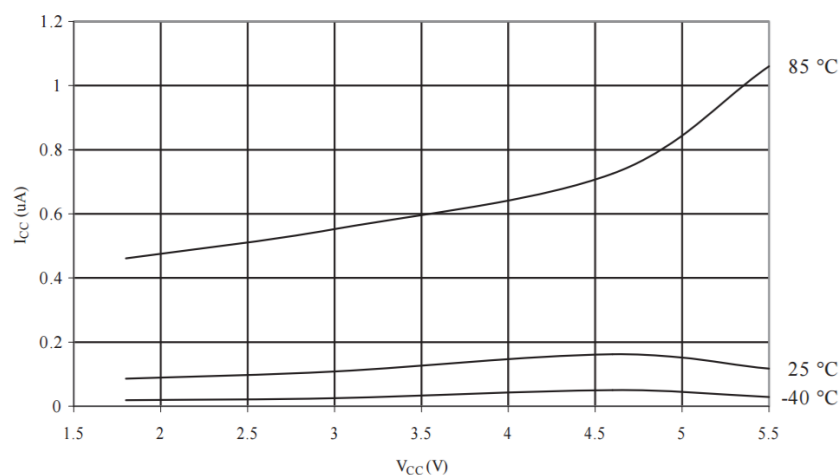
### 2.1 picoPower

Jedným z cieľov tejto práce je navrhnuť bezdrôtový senzor, ktorý bude mať veľmi malú spotrebu energie, preto je jasnou voľbou verzia picoPower, ktorá by mala zvýšiť životnosť celého senzorového zariadenia. Táto verzia zahŕňa:



- možnosť vypnutia alebo zníženie taktovacej frekvencie častí zariadení, ktoré sa nevyužívajú
- technológiu „SleepWalking“, ktorá umožní procesoru prejsť do hlbokého spánku a zobudí ho iba v prípade, kedy je to naozaj nutné. Procesor sa tak nemusí v pravidelných intervaloch spúšťať do aktívneho režimu a vyhodnocovať situáciu
- rýchlejší prechod do aktívneho režimu
- nižšia spotreba pri zachovaní plnej funkcionality [8] [9]

Vhodnou kombináciou uspávania sme by sme mali byť teoreticky schopní sa dostať sa pri teplote 25 °C a napájacom napätí 3 V na hodnotu prúdu pohybujúcu sa okolo 100 nA.



Obr. 2.2: Spotreba prúdu mikrokontrolérom pri rôznom napájacom napätí a teplote

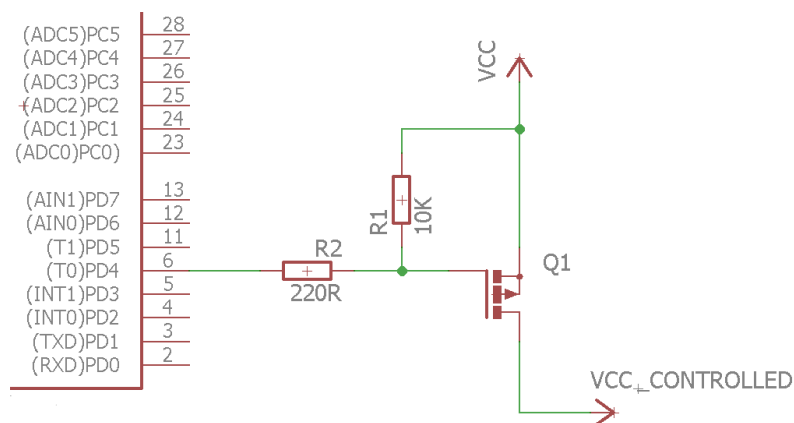
Aby sme sa čo najviac priblížili k minimálnej spotrebe a mohli naše zariadenie poháňať na batérie čo najdlhšie, budeme sa snažiť dodržiavať tieto zásady:

- využívať mikrokontrolér na čo najnižšej frekvencii a napätí
- uspať mikrokontroléra pomocou rôznych režimov spánku
- softvérovo vypínať nepotrebné vstavané moduly (napr. AD prevodník, SPI...)
- vypínať zisťovanie trvalého podpätia (BOD) pri režime spánku
- nepoužívať LED diódy signalizujúce zapnutie prístroja
- odpojiť externé súčasti v čase ich nevyužitia (napr. s tranzistorom MOSFET)

## 2.2 Vypínanie periférii

Životnosť batérii nášho zariadenia chceme čo najviac predĺžiť, preto musíme obmedziť aj spotrebu externých zariadení, nenechať ich v režime, kedy by pasívne mŕňali energiu.

Najlepšie sa zdá byť riešenie, kedy priamo odpojíme napájanie daných externých súčastí. Avšak možnosti, ktoré by vyžadovali zásah užívateľa nie sú vhodné. Preto by bolo vhodné využiť „softvérový vypínač“ v podobe tranzistoru P kanálový MOSFET. Aby sme ho zopli a teda priviedli do periférie napájanie, bránu tranzistora nastavíme na 0 V.



Obr. 2.3: Zapojenie tranzistora MOSFET pre ovládanie napájania periférií

Rezistor s hodnotou 220 Ω je využitý pre ochranu mikrokontroléra a obmedzuje prúd na 15 mA. Aby sme teda spustili napájanie, keď sa zariadenie zobudí, využijeme zmenu stavu pinu na LOW. [10]

## 3 ODOSIELANIE DÁT S ATMEGA328P

### 3.1 XBee

XBee je veľmi populárne riešenie bezdrôtovej komunikácie zariadení, prevažne súkromných sietí, vyvinuté spoločnosťou Digi International. Tento modul využíva komunikačný protokol IEEE 802.15.4 / ZigBee a jedna z jeho výhod je nízka spotreba energie. Dosah zariadenia je v rozpätí 10 až 100 metrov v závislosti na charakteristike prostredia. Silnou stránkou je však doručenie dát pomocou väčších zložených sietí. S Arduino komunikuje pomocou rozhrania UART.



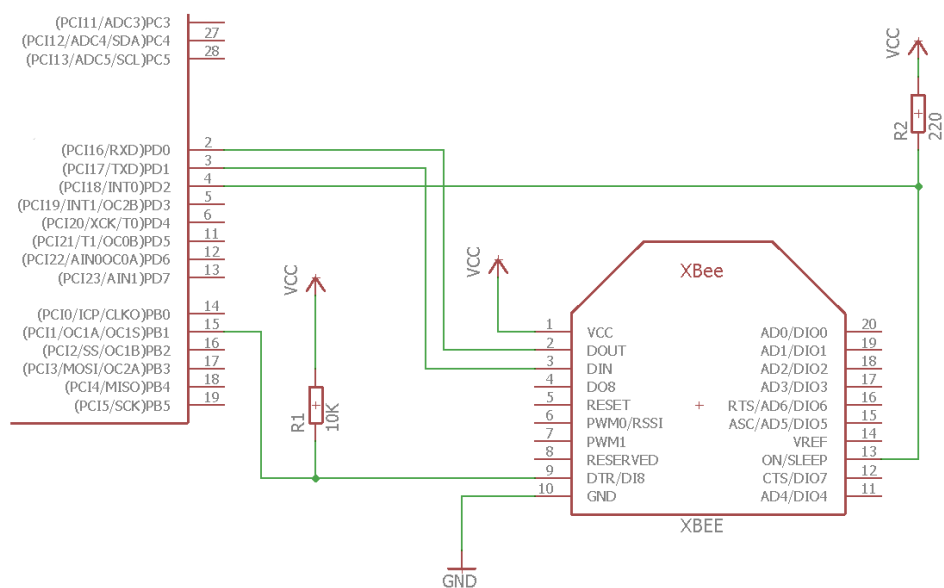
Obr. 3.1: Modul XBee [11]

XBee modul môže byť nastavený v dvoch režimoch - v jednoduchom a aplikačnom (API). V jednoduchom móde sú dáta privádzané na pin DIN priamo vysielané až do prijímacieho zariadenia bez akejkoľvek modifikácie. Táto funkcia je bezdrôtovým ekvivalentom ku štandardu káblu RS-232. Pakety môžu byť smerované na jeden konkrétny cieľ, alebo vysielané do viacerých zariadení súčasne. [12] [13]

#### 3.1.1 Uspávanie XBee

Aj zariadenie XBee nám ponúka možnosť spánku a režimy nízkej energie. My využijeme najhlbší režim spánku, ktorý sa nazýva Pin Hibernation. Tento režim minimalizuje spotrebu energie. Aby sme nastavili XBee ku hibernácii, musíme správne nastaviť parametre SM a D7. SM parameter konkrétne nastavuje režim Pin Hibernation a parameter D7 zakáže možnosť prerušenia vysielania dát. To vykonáme cez program XCTU, ktorý je určený na konfiguráciu XBee. [14]

Po nakonfigurovaní prechod zariadenia do režimu hibernácie vykonáme tak, že na DTR pin (na XBee je to pin číslo 9) pripojíme napätie 3,3 V. Ak na pin neprivedieme žiadne napätie, zostane 0 V, XBee sa prebudí.



Obr. 3.2: Zapojenie modulu XBee s funkciou uspávania

Aby sme teda XBee uspali, musí byť pin 9 (PB1) na mikrokontroléri ATmega328P nastavený ako vstup a nastavený ako HIGH. Keď budeme potrebovať XBee zobudiť, pin na mikrokontroléri nastavíme ako výstup LOW, čo stiahne napájanie na XBee pine DTR a uskutoční sa požadované vzбудenie.

Na to, aby sme bezpečne odoslali a prijali dáta, vytvoríme si akýsi paket.

Hlavička	ID	Dáta	Kontrolná suma
----------	----	------	----------------

Obr. 3.3: Paket odosielaný XBee

Na začiatku je nejaká fixne daná hodnota, ktorá bude značiť začiatok paketu, napríklad hexadecimálna hodnota 0xFF teda 255. Následne senzorový uzol oznámi prijímateľovi (bráne) svoju identitu. Ďalšou časťou sú priamo údaje z meraných senzorov, ktoré sú nasledované kontrolnou sumou, ktorá slúži na overenie chýb v prenose a je to hodnota zložená z ID a hodnoty teploty. [15]

```

1
2 int XBee_wake = 9;
3
4 void setup(void) {}
5
6 void loop(void) {
7
8 // zobudenie XBee
9 pinMode(XBee_wake, OUTPUT);
10 digitalWrite(XBee_wake, LOW);
11
12 // uspanie XBee
13 pinMode(XBee_wake, INPUT);
14 digitalWrite(XBee_wake, HIGH);
15
16 }
17

```

Obr. 3.4: Príklad kódu pre uspanie a zobudenie XBee

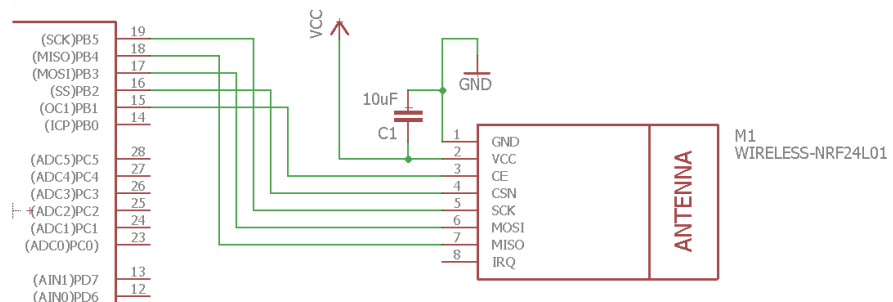
## 3.2 nRF24L01 2.4GHz

Bezdrôtová komunikácia medzi dvoma a viacerými Arduinami nám ponúka nové možnosti. Jedným využitím bude aj naše odosielanie údajov pre monitorovanie výstupu zo senzorov. Na svete však existuje množstvo nízkonákladových 2.4 GHz rádio modulov založených práve na spomínanom Nordic Semiconductor nRF24L01 module.

### 3.2.1 Dosah

Dosah tohto zariadenia je veľmi závislý na situácii a podmienkach, v ktorých sa nachádza. V externých podmienkach pri priamej viditeľnosti je omnoho vyšší ako v interiéri s množstvom predeľovacích stien a vybavenia. Podľa výrobcu je pre nízkopríkonovú verziu modulu predpokladaný dosah v otvorenom priestore približne 100 metrov.

Užitočnou radou pre zlepšenie dosahu je nastavenie prenosovej frekvencie na hodnotu prevyšujúcu drvivú väčšinu bežne dostupných WiFi zariadení. Základný rozsah zariadenia je medzi 2400 MHz a 2525 Mhz, pričom rozdelenie tohto pásma je po 1 MHz, čo znamená, že máme k dispozícii 125 rôznych kanálov označených 0 až 124. [16]



Obr. 3.5: Zapojenie nRF24L01 s ATmega328P

## 4 REŽIMY ÚSPORY A USPANIA

Jednotlivé režimy spánku nám pomáhajú vypnúť nepotrebné moduly a teda ušetriť energiu. Existujú rôzne variácie týchto režimov pre optimálne nastavenie užívateľských potrieb. Konkrétnym nastavením a spôsobom uvedenia do jednotlivých režimov spánku sa budeme zaoberať v časti Meranie s ATmega328P.

Existuje teda 6 režimov: [7]

### 4.1 Idle Mode

Vstup do režimu Idle spôsobí zastavenie CPU, avšak ostatné súčasti остану v operačnom režime (napr. SPI, UART, WDT...). Zastavenie sa teda týka  $clk_{CPU}$  a  $clk_{FLASH}$  a chod ostatných hodín sa nezmení. Mikrokontrolér je možné prebudiť pokynom interným aj externým. Ak by sme chceli znížiť spotrebu v Idle režime a nepotrebujeme jeho prebudenia, analógový komparátor sa odporúča vypnúť.

### 4.2 ADC Noise Reduction Mode

Tento režim taktiež zastaví CPU,  $clk_{CPU}$  a  $clk_{FLASH}$ , ale navyiac je zastavené aj  $clk_{I/O}$ . Táto skutočnosť spôsobí zlepšenie prostredia pre AD prevodník, a tým pádom sú možné merania s väčšou presnosťou.

### 4.3 Power-Down Mode

V tomto režime sa zastaví externý oscilátor a jedinými možnosťami ako mikrokontrolér prebudiť je externý reset, Watchdog reset, Watchdog prerušenie, Brown-out reset, zhoda adres I<sup>2</sup>C, externé prerušenie na INT a zmena pinu. Tento spánok zastaví všetky hodiny, beh je umožnený len asynchrónnym modulom. Keďže tento režim ponúka najhlbší spánok, budeme ho využívať i my a nemôžeme zabúdať, že pri zobudení z tohto režimu je vhodné počítať s akýmsi oneskorením dokým nabehnú všetky súčasti.

### 4.4 Power-Save Mode

Podobnosť s Power-Down režimom je odlišená jedinou výnimkou a to, že Timer/Counter2 ostane bežať i počas spánku.

## **4.5. Standby Mode**

Opäť je tento režim takmer identický s Power-Down režimom, no oscilátor ostane zapnutý. Mikrokontrolér sa zobúda z tohto režimu po šiestich hodinových cykloch.

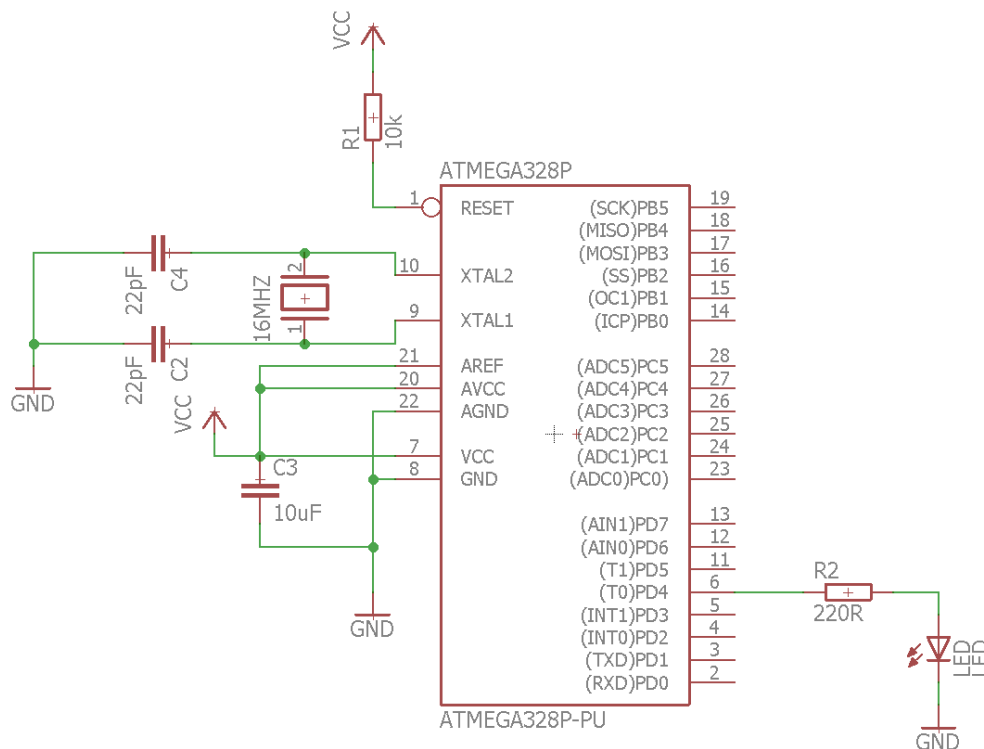
## **4.6. Extended Standby Mode**

Podľa katalógového listu je tento režim úplne rovnaký ako Standby režim a zariadenie sa taktiež zobúda po šiestich hodinových cykloch.

## 5 MERANIE S ATMEGA328P

Pre výber a otestovanie najvhodnejších techník pre náš projekt vytvoríme zapojenie s mikrokontrolérom ATmega328P, na ktorom jednoducho odmeriame jednotlivé možnosti úspory energie, ktorej pri napájaní batériami nikdy nie je nazvyš. Tieto možnosti sú plne uplatniteľné ako pre ATmega328P tak i ATmega256RFR2.

V tomto meraní budeme využívať hlboký spánok mikrokontroléra a následné „zobudenie“, teda privedenie do operačného režimu pomocou WDT. V tomto pokuse sa budem snažiť nevyužívať knižnice, ktoré sú však výborným nástrojom, ale všetky príkazy budú čerpané z popisu katalógové listu.



Obr. 5.1: Základné zapojenie mikrokontroléra ATmega328P

### 5.1 Režimy pre optimalizáciu spotreby

Najskôr otestujeme odporúčaných 5 V pre napájanie a 16 MHz kryštál. Pri jednoduchom kóde pre bliknutie LED diódy bola spotreba 24,98 mA pri zapnutí LED a 13,17 mA pri vypnutí.



```

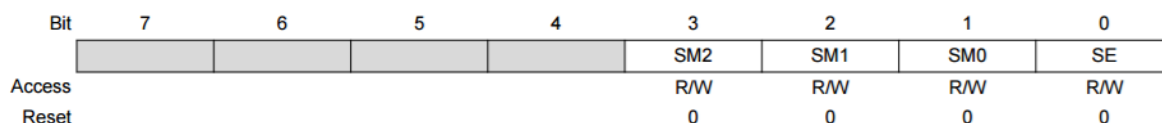
1 #define LED_PIN 4
2 void setup() {
3     pinMode(LED_PIN, OUTPUT);
4 }
5
6 void loop() {
7     |
8     digitalWrite(LED_PIN, HIGH);
9     delay(1000);
10    digitalWrite(LED_PIN, LOW);
11    delay(3000);
12 }

```

Obr. 5.2: Základný kód bliknutia LED diódy

### 5.1.1 Power-down režim

Avšak, ak chceme znížiť spotrebu nášho zariadenia, fázu *delay()*, kedy LED nesvieti môžeme nahradiť spánkom. Z viacerých typov spánku vyberieme práve najhlbší spánok (Power-down režim). Pre nastavenie tohto režimu spánku využijeme SMCR kontrolný register pre režimy spánku opísaný v katalógovom liste, pričom musíme nastaviť konkrétne bity na tabuľkou určenú hodnotu. [7]



Obr. 5.3: SMCR – kontrolný register

Tab. 2: Postupnosť bitov pre určenie režimu spánku

bity SM2, SM1, SM0	typ spánku
0 0 0	Idle režim
0 0 1	ADC redukcia šumu
0 1 0	Power-down režim
0 1 1	Power-save režim
1 1 0	Stand-by režim
1 1 1	Rozšírený Stand-by režim

Ako je z tabuľky zrejmé, bit SM1 prestavíme z pôvodnej hodnoty 0 na hodnotu 1. Bit SE musíme taktiež prestaviť na hodnotu 1 pre prechod do režimu spánku. Tento bit treba prepísať tesne pred pokynom uspania a zmeniť jeho hodnotu na pôvodnú 0 opäť tesne po zobudení mikrokontroléra.

Aby sme však vykonali pokyn uspania, musíme spustiť inštrukciu SLEEP v strojovom prekladači, ktorý spustíme pomocou príkazu `__asm__ __volatile__`. [17]

```

1 #define LED_PIN 4
2 void setup() {
3     pinMode(LED_PIN, OUTPUT);
4 }
5
6 void loop() {
7
8     digitalWrite(LED_PIN, HIGH);
9     delay(1000);
10    digitalWrite(LED_PIN, LOW);
11
12    SMCR |= (1<<2);
13    SMCR |= 1;
14    __asm__ __volatile__ ("sleep");
15 }
16

```

Obr. 5.4: Zdrojový kód s využitím hlbokého spánku

S využitím tohto spánku sa znížila spotreba na 280  $\mu\text{A}$  pri vypnutí LED, avšak problémom je, že zatiaľ nemáme ako tento režim prerušiť.

### 5.1.2 Vypnutie AD prevodníka

Ďalšou z možností ako znížiť spotrebu je vypnutie analógovo-digitálneho prevodníka. V katalógovom liste zistíme, že ak zmeníme hodnotu siedmeho bitu ADCSRA na hodnotu 0, vypneme prevodník. [18]

Bit	7	6	5	4	3	2	1	0
	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Obr. 5.5: Register ADCSRA

Využijeme k tomu príkaz: „, `ADCSRA &= ~(1<<7);`“, ktorý vyčistí siedmy bit registra a teda ho uvedie na hodnotu 0. Spotreba v režimu spánku nám klesla ešte na hodnotu 19,8  $\mu\text{A}$ .

### 5.1.3 Vypnutie BOD

Spotrebu môžeme znížiť aj vypnutím detekcie trvalého podpätia počas spánku. Túto funkciu ovláda MCU kontrolný register – MCUCR.

Bit	7	6	5	4	3	2	1	0
		BODS	BODSE	PUD			IVSEL	IVCE
Access		R/W	R/W	R/W			R/W	R/W
Reset		0	0	0			0	0

Obr. 5.6: Kontrolný register – MCUCR

Pre vypnutie tohto zisťovania musí byť bit BODS prepísaný na hodnotu 1. Avšak zapisovanie tohto bitu je kontrolované spúšťacím bitom BODSE. Preto ak chceme vypnúť BOD, musia byť oba bity – BODS aj BODSE – zapísané ako hodnota 1 najneskôr v rozmedzí štyroch cyklov, a následne bit BODS znova zapísaný ako 1 a BODSE spať na hodnotu 0.

Spôsob ako to dosiahnuť môže byť nasledovný: Keďže oba bity BODS i BODSE treba nastaviť vo veľmi krátkom rozmedzí, najlepšie by ich bolo nastaviť súčasne. To by sme mohli vykonať zapísaním hodnoty 3, čo je v preklade do binárnej sústavy 11, do piateho bitu, čím teda zapíšeme aj šiesty bit.

Následne v ďalšom príkaze zoberieme práve zapísaný register MCUCR a vyčistíme v ňom piaty bit (BODSE) – zapísaná bude hodnota 0 a opätovne zapíšeme hodnotu 1 do šiesteho bitu (BODS).

Po tomto úkone ostáva detekcia trvalého podpätia aktívna ešte 3 cykly hodín. Inštrukcia ku uspaniu musí byť vykonaná ešte počas tohto obdobia. [7]

```

1 #define LED_PIN 4
2 void setup() {
3     pinMode(LED_PIN, OUTPUT);
4 }
5
6 void loop() {
7
8     digitalWrite(LED_PIN, HIGH);
9     delay(1000);
10    digitalWrite(LED_PIN, LOW);
11
12    ADCSRA &= ~(1<<7);
13
14    SMCR |= (1<<2);
15    SMCR |= 1;
16
17    MCUCR |= (3 << 5);
18    MCUCR = (MCUCR & ~(1 << 5)) | (1 << 6);
19    __asm__ __volatile__ ("sleep");
20 }
21

```

Obr. 5.7: Zdrojový kód s použitím vypnutia AD prevodníka a BOD

Pri takto zvolenom kóde je spotreba nášho testovacieho modulu v režime spánku rovná 0,127  $\mu$ A.

### 5.1.4 Využitie WDT

Implementácia tohto časovača, ktorý je úplne nezávislý od systémových hodín, v tomto prípade 16 MHz a nezaručuje takú presnosť, nám pomôže ukončiť režim spánku – zobudiť zariadenie. Na jej ovládanie potrebujeme poznať register WDTCSR:

Bit	7	6	5	4	3	2	1	0
	WDIF	WDIE	WDP[3]	WDCE	WDE	WDP[2:0]		
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Obr. 5.8: Register WDTCSR

Na to, aby sme nastavili bity WDP, potrebujeme do bitu WDE zapísať hodnotu 1. Aby sme to mohli urobiť, treba v rovnakom čase však zapísať aj do bitu WDCE hodnotu 1. V nasledujúcich štyroch cykloch potom môžeme nastaviť požadovanú hodnotu pre bity WDP a WDE. Preto použijeme číslo 24, ktoré opäť po prevedení do binárnej sústavy naraz vpíše do tretieho a štvrtého bitu hodnotu 1. Následne zapíšeme hodnotu 33, ktorá nám podľa tabuľky udáva hodnotu cirka 8 sekúnd a zároveň aj vynulujeme bity WDCE a WDE. Ako posledné zapíšeme do bitu WDIE jednotku, čím aktivujeme mód následného zobudení.

Tab. 3: Kombinácie bitov pre rozdielne dĺžky intervalu časovača

WDP [3]	WDP [2]	WDP [1]	WDP [0]	Počet WDT cyklov	Dĺžka [ms]
0	0	0	0	2048	16
0	0	0	1	4096	32
0	0	1	0	8192	64
0	0	1	1	16384	128
0	1	0	0	32768	256
0	1	0	1	65536	512
0	1	1	0	131072	1024
0	1	1	1	262144	2048
1	0	0	0	524288	4096
1	0	0	1	1048576	8192

Nám však v našom projekte nestačí spánok 8 sekúnd, potrebujeme, aby zariadenie vykonalo svoju činnosť raz za 10 minút. Preto použijeme pokyn uspania v slučke vždy

po 8 sekúnd až do uplynutia zvoleného časového obdobia. Musíme však pamätať na to, že je späť s vypínaním BOD. [7]

Keď zasiahne WDT, musíme pre fungovanie časovača použiť príkaz na jeho reset.

V tomto režime sa nám spotreba zvýšila, kvôli fungovaniu WDT, a to na 6,03  $\mu\text{A}$ .

Ďalšou časťou nášho merania bude zmena napájania na 3,3 V a následne kryštálu na 8 MHz hodnotu.

```
1 #define LED_PIN 4
2 void setup() {
3     pinMode(LED_PIN, OUTPUT);
4 }
5
6 void loop() {
7
8     digitalWrite(LED_PIN, HIGH);
9     delay(1000);
10    digitalWrite(LED_PIN, LOW);
11
12    WDTCSR = (24);
13    WDTCSR = (33);
14    WDTCSR |= (1<<6);
15
16    ADCSRA &= ~(1<<7);
17
18    SMCR |= (1<<2);
19    SMCR |= 1;
20
21    for(int x=0; x<75; x++){
22        MCUCR |= (3 << 5);
23        MCUCR = (MCUCR & ~(1 << 5)) | (1 << 6);
24        __asm__ __volatile__ ("sleep");
25    }
26 }
27
28 ISR(WDT_vect) {}
29
```

Obr. 5.9: Zdrojový kód s uplatnením vzbudenia WDT

Tab. 4: Rôzne možnosti napájanie a ich spotreba

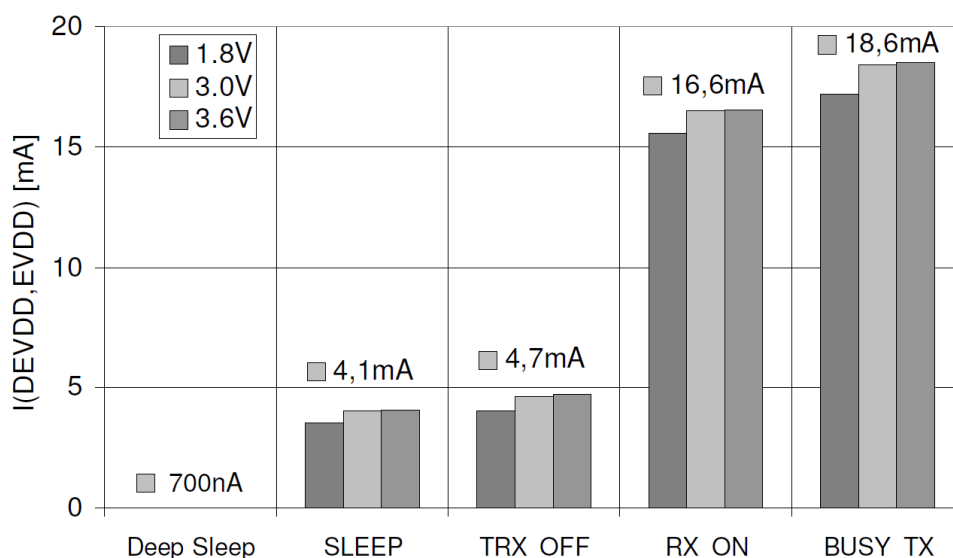
	Chod [mA]	Spánok [ $\mu\text{A}$ ]
5 V, 16 MHz	24,2	6,02
5 V, 8 MHz	20,8	6,03
3 V, 8 MHz	9,4	4,43

## 6 MIKROKONTROLÉR ATMEL

### ATMEGA256RFR2

Postupne ako sme sa dostávali hlbšie do tejto problematiky, naskytlo sa ďalšie, nové riešenie návrhu fungovania senzorového uzla. Najúčinnnejšou variantou by bolo predsa spojenie mikrokontroléra a zariadenia pre bezdrôtový prenos, nie dve rozdielne entity ako ATmega328P a XBee. Preto využijeme integrované riešenie v podobe osembitového mikrokontroléra taktiež od spoločnosti Atmel, model ATmega256RFR2. Tento optimalizovaný model spája všetky požiadavky, ktoré sme mali na ATmega328P vrátane zníženej spotreby energie a pridáva k týmto funkciám 2.4 GHz vysielateľ-prijímač pracujúci na linkovej vrstve 802.15.4, ktorý je jeden z najvýkonnejších vo svojej kategórii. Flash pamäť má maximálnu veľkosť 256 kB a EEPROM pamäť 8 kB. Pre väčšie uplatnenie do nášho návrhu zahrnieme aj rozšírenie týchto dvoch pamätí.

Štandard IEEE 802.15.4 sústreďuje technológie pre komunikáciu bezdrôtových sietí súkromného charakteru s nízkou prenosovou rýchlosťou, teda Low Rate Wireless Personal Area Network – LR-WPAN a špecifikuje parametre fyzickej a spojovej vrstvy pre vhodné fungovanie bezdrôtového zariadenia. [19]



Obr. 6.1: Spotreba mikrokontroléra a bezdrôtovej časti ATmega256RFR2 [20]

Z grafu na Obr. 1.12 môžeme odčítať typické prúdy napájania mikrokontroléra so 16 MHz oscilátorom a bezdrôtovej časti v najtypickejších situáciách. Pokiaľ bezdrôtovú časť uspíme, zaznamenáme zníženie prevádzkového prúdu o 13,9 mA oproti vysielaniu,

respektíve 11,9 mA oproti stavu prijímania dát. Tradične najmenej náročný na spotrebu je režim hlbokého spánku, kedy sú všetky hlavné digitálne súčasti odpojené od hlavného napájania. [20]

## 6.1. Atmel Lightweight Mesh

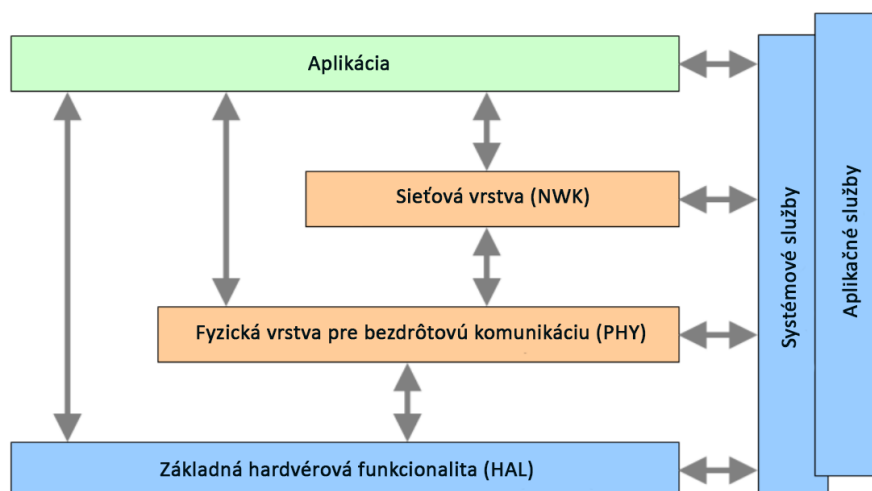
Lightweight Mesh je proprietárny bezdrôtový sieťový protokol firmy Atmel s dôrazom na nízku spotrebu energie a jednoduchosť použitia. Bol navrhnutý pre použitie v širokej škále uplatnení bezdrôtových technológií, ako napríklad:

- bezdrôtové ovládanie
- alarmy a zabezpečovanie
- automatické bezdrôtové vyčítavanie údajov
- automatizácia budov a domácností
- vybavenie pre výučbu a hračky

Kompatibilita tohto protokolu je naprieč celým spektrom Atmel mikrokontrolérov a teda aj s našim ATmega256RFR2. [21]

Pomocou Lightweight Mesh protokolu je možné vytvárať siete až o 65535 uzloch, avšak my v našom projekte využijeme jednu zo základných možností, a to model Peer-to-Peer.

Architektúra Lightweight Mesh je rozdelená do rôznych úrovní poskytujúc rozhrania vhodné pre užívateľa a obsahuje iba funkcie zabezpečujúce nevyhnutný chod bezdrôtovej komunikácie.



Obr. 6.2: Architektúra Lightweight Mesh [21]

Ako už z názvu vyplýva, vrstva HAL poskytuje základnú hardvérovú funkcionálnu ako hardvérový časovač, GPIO prístup k bezdrôtovému rozhraniu alebo kontrolu spánku.

Fyzická vrstva pre bezdrôtovú komunikáciu (PHY) poskytuje funkcie spojené s bezdrôtovým prenosom informácií, ale niektoré sú však prístupné len pre sieťovú vrstvu (napr. žiadosť o odoslanie dát) a niektoré sú použiteľné len z aplikácie (napr. výber frekvencie). [21]

Sieťová vrstva poskytuje jadro komunikácie zložené z množstva polí a rámcov.

Systémové služby spájajú vrstvy poskytovaním funkcií potrebných pre všetky z nich ako definície, konfiguračné parametre atď.

Aplikačné služby nie sú nevyhnutnou súčasťou, ale sú zväčša vyžadované aplikáciami. Momentálne jedinou službou, ktorá ich využíva je OTA aktualizácia. [21]

Pre programovanie nášho mikrokontroléra a editáciu zdrojového kódu použijeme program Atmel Studio, ktorý predstavuje vhodnú platformu pre prácu s 8-bitovými mikrokontrolérmi spoločnosti Atmel a je postavený na známom prostredí Microsoft Visual Studio 2015.

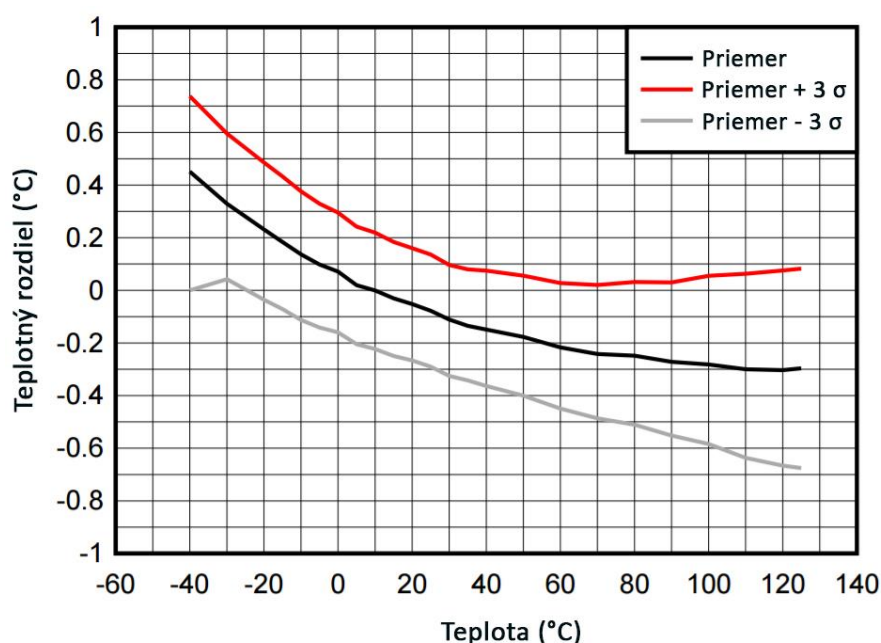


## 7 SENZORY S ATMEGA256RFR2

### 7.1 Teplotný senzor TMP102

Senzor, pomocou ktorého budeme zisťovať teplotu okolia je teplotný senzor TMP102AIDRLT v puzdre SOT563 od spoločnosti Texas Instruments. Tento senzor je digitálny, s typickou presnosťou  $\pm 0,5$  °C pri teplotách medzi -25 °C a 85 °C bez nutnosti kalibrácie alebo iných externých zariadení. Maximálne spektrum použiteľnosti je od -40 °C do 125 °C. [22]

Ďalšou jeho prednosťou je aj nízka spotreba energie, maximálny prúd v aktívnom režime činí 10  $\mu$ A počas celého teplotného funkčného spektra a v režime vypnutia senzora je to 1  $\mu$ A. Vstavaný je tiež aj 12 bitový AD prevodník s presnosťou na 0,0625 °C. Dodatočne využiteľnou vlastnosťou je aj režim prerušenia, ktorý využíva ALERT pin (v našom návrhu je samostatne vyvedený JP1) v prípade, ak teplota prekročí  $T_{(HIGH)}$  alebo klesne pod  $T_{(LOW)}$ . [22]



Obr. 7.1: Odchýlka teploty v merateľnom spektre [22]

## 7.2 Svetelný senzor ISL29020

Svetelný senzor ISL29020IROZ-T7 od firmy Intersil budeme v našom návrhu využívať pre jeho nízku spotrebu energie – maximálny spotrebovaný prúd 65  $\mu\text{A}$ , typicky 55  $\mu\text{A}$  v zaťažení a 0,5  $\mu\text{A}$  v režime vypnutia. Veľkou výhodou je, že tento svetelný senzor dokáže vylúčiť z meraného spektra ako infračervené, tak aj ultrafialové žiarenie a pre svoje fungovanie taktiež nepotrebuje žiadne komplexné algoritmy. Rozlíšenie AD prevodníka sa pohybuje až na úrovni 16 bitov a na komunikáciu budeme využívať jeho I<sup>2</sup>C rozhranie. [23]

Najrozšírenejšie použitia tohto senzora sú:

- Úprava jasu displeja pre mobilné zariadenia, počítače a inú elektroniku
- Zisťovanie mierky osvetlenia v lekárskom a priemyselnom odvetví

## **8 CENA ZARIADENIA S ATMEGA256RFR2**

Pre vyčíslenie ceny zariadenia sme použili internetový obchod [www.cz.farnell.com](http://www.cz.farnell.com). V našej cenovej kalkulácii zanedbávame predajcom minimálne množstvá odberu, ktoré uplatňuje na väčšinu súčiastok a budeme sa snažiť spočítať jednotkovú cenu pre zariadenie. Samozrejme, množstvo súčiastok môže byť použitých od iného výrobcu, prípadne dodávateľa, preto je táto suma len pre akúsi základnú predstavu finančnej náročnosti tohto návrhu.

Tab. 5: Cenová kalkulácia zariadenia

Počet kusov	Názov	Jednotková cena	Cena spolu
1 ks	8-bitový mikrokontrolér ATMEGA256RFR2-ZU	172,82 CZK	172,82 CZK
1 ks	Svetelný senzor ISL29020IROZ-T7	104,35 CZK	104,35 CZK
1 ks	Tepelný senzor TMP102AIDRLT	61,56 CZK	61,56 CZK
1 ks	MOSFET tranzistor, P kanál IRLML6402GTRPBF	6,96 CZK	6,96 CZK
1 ks	Pevný LDO napäťový regulátor, MCP1802T-3002I/OT	14,25 CZK	14,25 CZK
1 ks	16 MHz kryštál, SMD MCRSD16000F183000RR	5,30 CZK	5,30 CZK
1 ks	32.768 kHz kryštál LFX TAL002997	6,16 CZK	6,16 CZK
1 ks	Transformátor pre Baluny 2450FB15L0001E	28,16 CZK	28,16 CZK
1 ks	RF / Koaxiálny konektor U.FL-R-SMT-1	17,95 CZK	17,95 CZK
2 ks	22 pF kondenzátor SMD 06035A220JAT2A	0,53 CZK	1,07 CZK
10 ks	0.1 $\mu$ F kondenzátor SMD C0603C104K4RACTU	0,41 CZK	4,14 CZK
2 ks	4,7 pF kondenzátor SMD 251R14S4R7BV4T	3,70 CZK	7,40 CZK
2 ks	1 pF kondenzátor SMD MC0603N1R0C500CT	0,33 CZK	0,67 CZK
2 ks	1 $\mu$ F kondenzátor SMD 0805B105K160CT	1,68 CZK	3,36 CZK
2 ks	10 k $\Omega$ rezistor SMD CRCW060310K0FKEA	0,46 CZK	0,93 CZK
2 ks	1 k $\Omega$ rezistor SMD CRCW06031K00FKEA	0,46 CZK	0,93 CZK
1 ks	0 $\Omega$ rezistor SMD MC0603SAF0000T5E	0,27 CZK	0,53 CZK
1 ks	499 $\Omega$ rezistor SMD CRCW0603499RFKEA	0,46 CZK	0,93 CZK
2 ks	LED zelená SMD KPT-1608SGC	1,57 CZK	3,15 CZK
1 ks	LED červená SMD KPT-1608EC	1,77 CZK	1,77 CZK
<b>SPOLU</b>		<b>442,36 CZK</b>	

# ZÁVER

V tejto bakalárskej práci sme sa pokúsili ozrejmiť jednotlivé časti navrhovaného nízkopríkonového bezdrôtového uzla, zapojenia, ako aj spôsoby ako eliminovať spotrebu a predĺžiť tak životnosť batérií, kedy sa ako najdôležitejšie javia prechody do režimov spánku a vypínanie súčastí.

Po porovnaní dvoch variant – jedenej s ATmega328P a druhej ATmega256RFR2, je vhodnejším kandidátom práve ATmega256RFR2. Veľkosť dosky je nezanedbateľne menšia, vzhľadom na komplexnosť riešenia – pri návrhu s ATmega328P by sme museli jednotlivé komponenty na seba vrstviť, napríklad modul XBee. Pri rozhodovaní o cene zariadenia je ATmega256RFR2 taktiež jasným víťazom. Cena tohto mikrokontroléra je už spomenutých 172,82 CZK, zatiaľ čo pre nás významovo rovnako funkčné duo zariadení ATmega328P a XBee sa pohybuje na úrovni cirka 750 CZK. Ďalšou kategóriou, v ktorej je výhodnejšia ATmega256RFR2 je spotreba. Kým spojenie ATmega328P a XBee v operačnom režime má spotrebu približne 60 mA, pri optimalizovanej verzii senzorového uzla s mikrokontrolérom ATmega256RFR2 je to len tretina.

Náš finálny návrh pozostáva z upravenej verzie Atmel Xplained Pro, čo je výborný nástroj pre vývoj a testovanie možností mikrokontroléra ATmega256RFR2 oficiálne podporovaný mnohými dokumentami a návodmi pre rôzne návrhy využitia priamo od spoločnosti Atmel známe pod názvom Atmel Software Framework. Varianty použitia je taktiež možné rozšíriť pomocou množstva doplnkových modulov, ktorých je neúrekom.

O snímanie dát v našom zariadení sa stará dvojica spomenutých senzorov – cez rozhranie I<sup>2</sup>C teplotu sníma TMP102AIDRLT a osvetlenie ISL29020IROZ-T7. Na softvérovú komunikáciu sme využili aj príklady použitia zariadení vyvinuté firmou Dresden Elektronik, ktorými sú napríklad ich moduly deRFmega a zariadenie deRFnode. [24]

Po tom ako senzory vyčítajú údaje z prostredia, sú namerané dáta odoslané na zbernú stanicu. Pre danú komunikáciu používame upravenú verziu Peer-to-Peer, čo je príklad komunikácie poskytnutý v Lightweight Mesh Software Stack, ktorý je voľne stiahnuteľný na stránkach spoločnosti Atmel a ponúka aj množstvo iných návrhov na využitie bezdrôtovej komunikácie možnej mikrokontrolérom ATmega256RFR2.

Napájanie nášho zariadenia je zabezpečené troma vymeniteľnými batérami štandardnej veľkosti AA. Takýto zdroj napájania by mal poskytovať dostatočné množstvo energie pre dlhodobé fungovanie zariadenia a ponúka aj istý komfort používateľovi. Tento komfort spočíva napríklad v tom, že nemusí riešiť umiestnenie senzorového uzla na svetelnom mieste, kvôli dobíjaniu energie cez solárne články. V pôvodnom návrhu figuroval kvôli zníženiu rozmerov zariadenia jeden dobíjateľný akumulátor LS 14500, čo je lítium thionyl-chloridová batéria ( $\text{LiSOCl}_2$ ) schopná pracovať bez problémov v silných mrazoch (až do  $-55^\circ\text{C}$ ) s jednou z najvyšších energických hustôt. Špeciálne sa hodí pre zariadenia s malými nárokmi na odoberaný prúd a jej životnosť je viac ako 10 rokov. Nevýhodou by však bola výmena tohto akumulátora prípadne jeho spomínané dobíjanie. [25]

Pre rozšírenie možností použitia nášho zariadenia je po okrajoch dosky navrhnuté rozloženie pinov v štýle Arduina. Toto rozšírenie je zahrnuté na úkor minimalizácie senzorového uzla, ale jeho prínos môže signifikantne rozšíriť použiteľnosť. Voliteľnou súčasťou je taktiež dopĺňajúca FLASH a EEPROM pamäť, ktorá môže byť doplnená do zariadenia.

# LITERATÚRA

- [1] WILLIAM STEWART, *The Internet Toaster* [online]. 07.01.2000 [cit. 18.10.2016]. Dostupné: <[http://www.livinginternet.com/i/ia\\_myths\\_toast.htm](http://www.livinginternet.com/i/ia_myths_toast.htm)>
- [2] KNUD LASSE LUETH, *Why the Internet of Things is called Internet of Things: Definition, history, disambiguation* [online]. 19.12.2014 [cit. 20.10.2016]. Dostupné: <<https://iot-analytics.com/internet-of-things-definition/>>
- [3] GOOGLE TRENDS, [online]. 11.11.2016 [cit. 11.11.2016]. Dostupné: <<https://www.google.sk/trends/explore?date=all&q=internet%20of%20things>>
- [4] POSTSCAPES, *Internet of Things History* [online]. 01.02.2016 [cit. 11.11.2016]. Dostupné: <<http://www.postscapes.com/internet-of-things-history/>>
- [5] Arduino UNO [online]. [cit. 21.11.2016]. Dostupné: <<https://www.arduino.cc/en/Main/ArduinoBoardUno>>
- [6] MATE MARSCHALKO, *Arduino Cheat Sheet* [online]. [cit. 22.11.2016]. Dostupné: <<http://www.webondevices.com/resources/arduino-cheat-sheet/>>
- [7] ATMEL, *ATmega328P Datasheet Complete* [online]. 11.2016 [cit. 22.11.2016]. Dostupné: <[http://www.atmel.com/Images/Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P\\_Datasheet.pdf](http://www.atmel.com/Images/Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P_Datasheet.pdf)>
- [8] ATMEL, *Low Power* [online]. [cit. 25.11.2016]. Dostupné: <<http://www.atmel.com/Technologies/lowpower/default.aspx>>
- [9] ATMEL, *picoPower Technology* [online]. [cit. 25.11.2016]. Dostupné: <<http://www.atmel.com/technologies/lowpower/picopower.aspx>>
- [10] NICK GAMMON, *Temperature and humidity sensor - battery powered* [online]. 05.08.2013 [cit. 25.11.2016]. Dostupné: <<http://www.gammon.com.au/forum/?id=12106>>
- [11] SPARKFUN, *Exploring XBees and XCTU* [online]. [cit. 30.11.2016]. Dostupné: <[https://cdn.sparkfun.com/assets/learn\\_tutorials/2/2/3/xbee-wire.jpg](https://cdn.sparkfun.com/assets/learn_tutorials/2/2/3/xbee-wire.jpg)>
- [12] DIGI, *XBEE/RF SOLUTIONS* [online]. [cit. 01.12.2016]. Dostupné: <<https://www.digi.com/products/xbee-rf-solutions/rf-modules/xbee-zigbee>>

- [13] JANAKIRAM MSV, *Tutorial: Configuring a Sensor Node and IoT Gateway to Collect and Visualize Data — Part 2* [online]. 06.07.2015 [cit. 28.11.2016]. Dostupné: <<http://thenewstack.io/tutorial-configuring-a-sensor-node-and-iot-gateway-to-collect-and-visualize-data-part-2/>>
- [14] *Low-Power Xbee Sleep Mode With Arduino And Pin Hibernation* [online]. 18.11.2012 [cit. 03.12.2016]. Dostupné: <<http://www.fiz-ix.com/2012/11/low-power-xbee-sleep-mode-with-arduino-and-pin-hibernation/>>
- [15] BERK SOYSAL, *Wireless Sensor Networks Project with Arduino + XBee Modules* [online]. 20.07.2014 [cit. 17.12.2016]. Dostupné: <<http://www.codemio.com/2014/07/wireless-sensor-networks-project-with.html>>
- [16] NORDIC SEMICONDUCTOR, *nRF24L01* [online]. [cit. 24.11.2016]. Dostupné: <<http://www.nordicsemi.com/eng/Products/2.4GHz-RF/nRF24L01/>>
- [17] HAN-WAY HUANG: *The Atmel AVR Microcontroller: MEGA and XMEGA in Assembly and C*, vyd. Delmar, Cengage Learning, 2013 816 s. ISBN-13:978-1-133-60729-8
- [18] PETER MORTENSEN, *How do you set, clear and toggle a single bit in C/C++?* [online]. 22.05.2015 [cit. 04.12.2016]. Dostupné: <<http://stackoverflow.com/questions/47981/how-do-you-set-clear-and-toggle-a-single-bit-in-c-c>>
- [19] *IEEE 802.15.4* [online]. [cit. 22.04.2017]. Dostupné: <[https://en.wikipedia.org/wiki/IEEE\\_802.15.4](https://en.wikipedia.org/wiki/IEEE_802.15.4)>
- [20] ATMEL, *ATmega256RFR2 Datasheet Complete* [online]. 09.2014 [cit. 02.05.2017]. Dostupné: [http://www.atmel.com/Images/Atmel-8393-MCU\\_Wireless-ATmega256RFR2-ATmega128RFR2-ATmega64RFR2\\_Datasheet.pdf](http://www.atmel.com/Images/Atmel-8393-MCU_Wireless-ATmega256RFR2-ATmega128RFR2-ATmega64RFR2_Datasheet.pdf)
- [21] ATMEL, *AVR2130: Lightweight Mesh Developer Guide* [online]. 03.2014 [cit. 11.05.2017]. Dostupné: <[http://www.atmel.com/Images/Atmel-42028-Lightweight-Mesh-Developer-Guide\\_Application-Note\\_AVR2130.pdf](http://www.atmel.com/Images/Atmel-42028-Lightweight-Mesh-Developer-Guide_Application-Note_AVR2130.pdf)>



- [22] TEXAS INSTRUMENTS, *TMP102 Low-Power Digital Temperature Sensor With SMBus and Two-Wire Serial Interface in SOT563* [online]. 12.2015 [cit. 16.05.2017]. Dostupné: <<http://www.ti.com/lit/ds/symlink/tmp102.pdf>>
- [23] INTERSIL, *ISL29020 Data Sheet* [online]. 20.08.2009 [cit. 18.05.2017]. Dostupné: <<http://www.intersil.com/content/dam/Intersil/documents/isl2/isl29020.pdf>>
- [24] Dresden Elektronik, *deRFnative Examples* [online]. [cit. 19.05.2017]. Dostupné: <<https://www.dresden-elektronik.de/funktechnik/products/software/sources-stacks-firmware/derfnative-examples/?L=1>>
- [25] SOS Electronic, *10 rokov prevádzky na 1 batériu?* [online]. 20.12.2013 [cit. 18.05.2017]. Dostupné: <<http://www.sos.sk/articles/xeno/xeno-10-rokov-prevadzky-na-1-bateriu-1463>>

# ZOZNAM SKRATIEK

API – Application Programming Interface

BOD – Brown-Out Detection

CE – Chip Enable

CSN – Chip Select Not

DIN – Data In

DMA – Direct Memory Access

GPIO – General-purpose input/output

IEEE – Institute of Electrical and Electronics Engineers

IoT – Internet of Things

IRQ – Interrupt Request

MISO – Master In Slave Out

MOSI – Master Output Slave Input

OTA – Over-The-Air

PWM – Pulse Width Modulation

RF – Radio Frequency

RFID – Radio Frequency Identification

SCK – Serial Clock

SMCR – Sleep Mode Control Register

SNMP MIB – Simple Networking Management Protocol Management Information Base

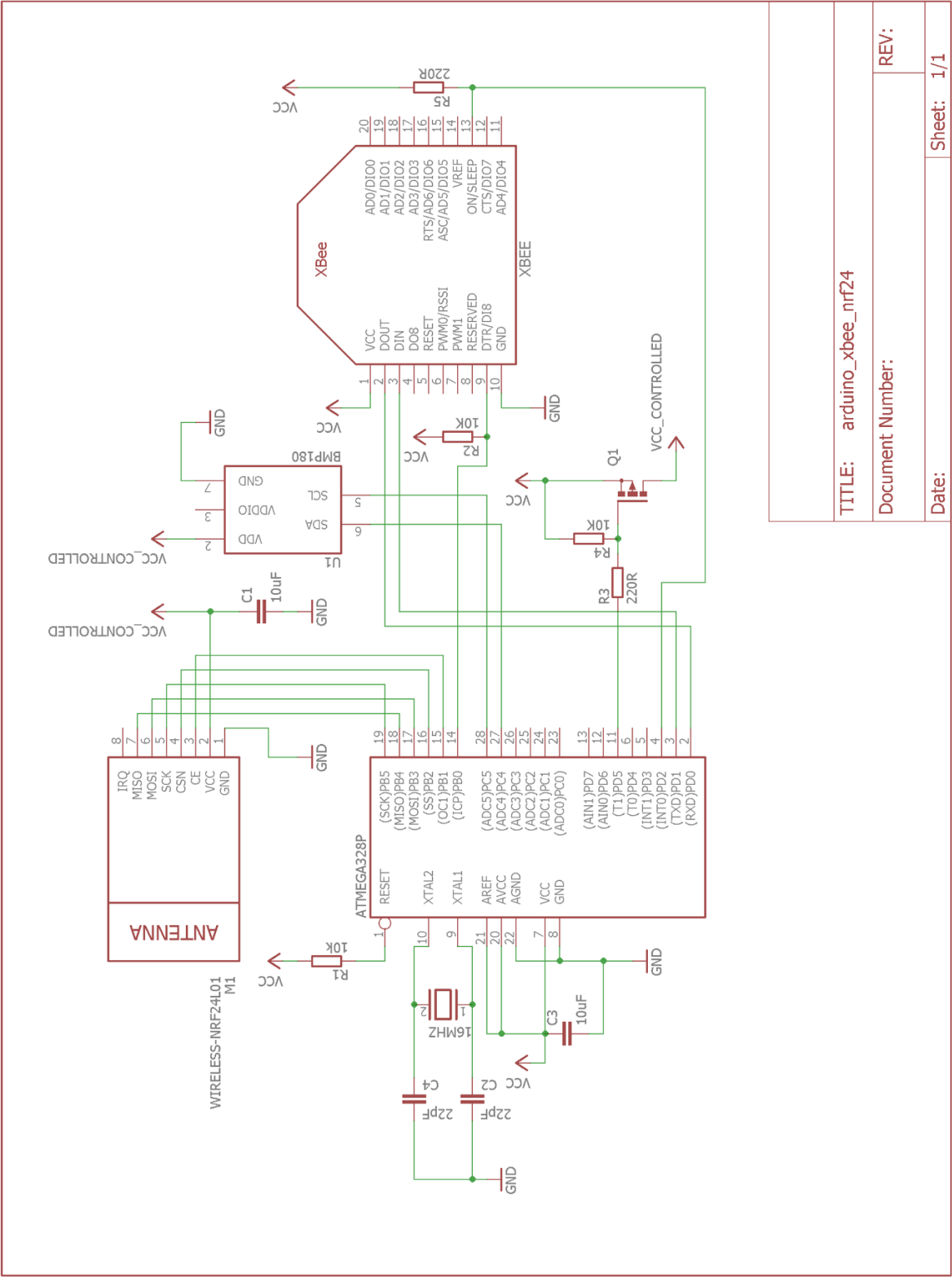
WDT – Watchdog Timer

# **ZOZNAM PRÍLOH**

- PRÍLOHA A:** Schematické zapojenie s ATmega328P
- PRÍLOHA B:** Schematické zapojenie s ATmega256RFR2
- PRÍLOHA C:** Návrh dosky plošného spoja senzorového uzla
- PRÍLOHA D:** Obsah CD

# PRÍLOHA A

## Schematické zapojenie s ATmega328P



TITLE: arduino\_xbee\_nrf24

Document Number:

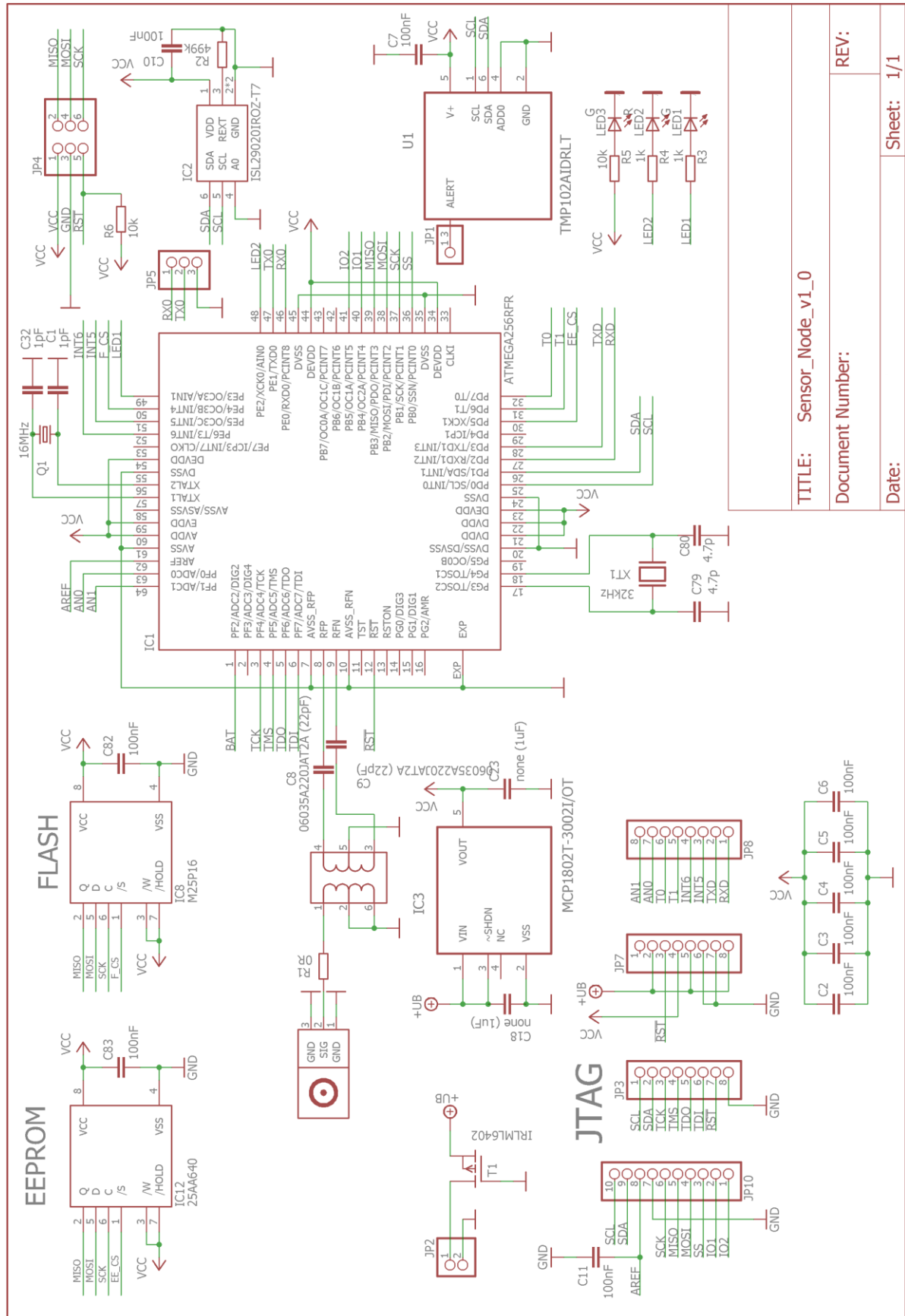
REV:

Date:

Sheet: 1/1

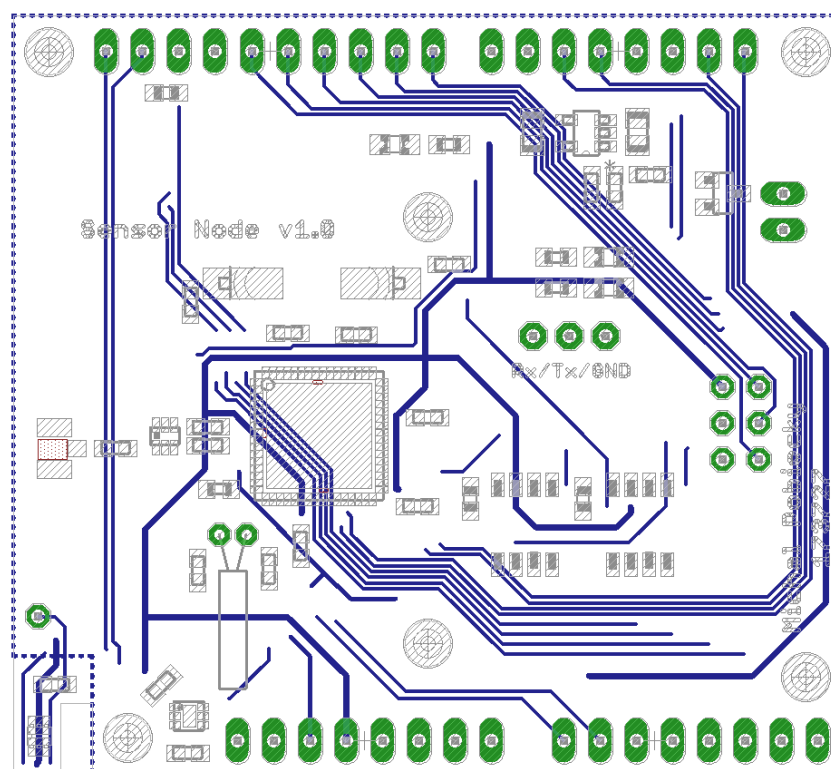
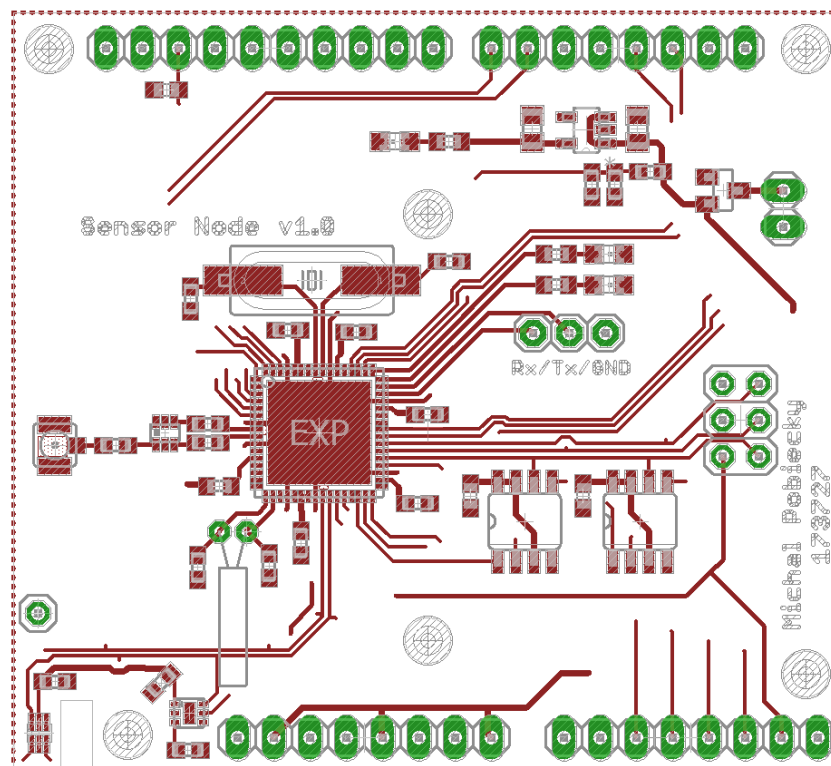
# PRÍLOHA B

## Schematické zapojenie s ATmega256RFR2



# PRÍLOHA C

## Návrh dosky plošného spoja senzorového uzla



# **PRÍLOHA D**

## **Obsah CD**

- Schéma a návrh dosky plošného spoja v programe EAGLE
- Zdrojový kód pre nízkopríkonový bezdrôtový senzorový uzol
- Elektronická verzia práce